FALL3D 5.0

Computer Code and Related Documentation

Giovanni Macedonio, Arnau Folch and Antonio Costa

Istituto Nazionale di Geofisica e Vulcanologia
Sezione "Osservatorio Vesuviano"
Via Diocleziano 326 -80124 Napoli, Italy

June 2007

Contents

1 Introduction **3**
1.1 Governing equation . . . 3
1.2 Parameterisation of the Eddy Diffusivity Tensor . . . 3
1.3 Settling velocity models . . . 4

2 Program setup **5**
2.1 Installation . . . 5
2.2 Folder structure . . . 6
2.3 Program run . . . 6

3 The FALL3D 5.0 input files **7**
3.1 The control file `FileInp` . . . 7
3.1.1 BLOCK TIME_UTC . . . 7
3.1.2 BLOCK FALL3D . . . 7
3.2 The source file `FileSrc` . . . 8
3.3 The granulometry file `FileGrn` . . . 9
3.4 The source file `FileDbs` . . . 9

4 The FALL3D 5.0 output files **9**
4.1 The list file `FileLst` . . . 9
4.2 The results file `FileRes` . . . 9

5 The utility programs **9**
5.1 The program SETGRN . . . 9
5.1.1 Program execution . . . 10
5.1.2 The GRANULOMETRY block . . . 10
5.2 The program SETSRC . . . 11
5.2.1 Program execution . . . 11
5.2.2 The SOURCE block . . . 12
5.3 The program BUILDDBS . . . 13
5.3.1 Program execution . . . 13
5.3.2 The METEO_DATABASE block . . . 13
5.3.3 The meteo data file `FileDat` . . . 14
5.3.4 The topography file `FileTop` . . . 14
5.4 The program MODELPOSTP . . . 15
5.4.1 Program execution . . . 15
5.4.2 The POSTPROCESS_MODELS block . . . 16

6 References **18**

1 Introduction

FALL3D 5.0 is a code written in FORTRAN90 which circumvents most of the simplifications behind the simpler volcanic ash fallout models. The model solves the time-dependent 3D advection-diffusion-sedimentation equation by means of a finite differences explicit scheme using a regular mesh. It uses the gradient transport theory to evaluate the atmospheric turbulent diffusion within and above the Atmospheric Boundary Layer (ABL) and experimental fits for the particle settling velocities, in addition to values from a dataset that contains full 3D prognostic wind field, source term, and topography. The model can, consequently, be used to forecast either ash concentration in the atmosphere or ash loading on the ground.

1.1 Governing equation

The non-conservative form of continuity equation written in a generalised coordinate system (X, Y, Z) is:

$$\begin{aligned} &\frac{\partial C}{\partial t} + U\frac{\partial C}{\partial X} + V\frac{\partial C}{\partial Y} + (W - V_{sj})\frac{\partial C}{\partial Z} = -C\nabla\cdot\mathbf{U} + C\frac{\partial V_{sj}}{\partial Z} \\ &\frac{\partial}{\partial X}\left(\rho_* K_X \frac{\partial C/\rho_*}{\partial X}\right) + \frac{\partial}{\partial Y}\left(\rho_* K_Y \frac{\partial C/\rho_*}{\partial Y}\right) + \frac{\partial}{\partial Z}\left(\rho_* K_Z \frac{\partial C/\rho_*}{\partial Z}\right) + S_* \end{aligned} \tag{1}$$

where C is the scaled average concentration, (U, V, W) are the scaled wind speeds, K_X, K_Y and K_Z are the diagonal scaled diffusion coefficients, ρ_* the scaled atmospheric density and S_* is the source term in the generalized coordinate system. Considering as a frame of reference a simple terrain-following coordinate system where the horizontal coordinates remain unchanged with respect to the Cartesian $(x = X,\ y = Y,\ z \to Z)$, the scaling factors are those reported in Table 1. Equation (1) is solved for each particle velocity class independently, i.e. assuming no interaction between particles belonging to different classes during the transport process. For a detailed description see Costa et al. (2005).

1.2 Parameterisation of the Eddy Diffusivity Tensor

In order to solve equation (1) it is necessary to evaluate the vertical and horizontal diffusion coefficients. Inside the atmospheric surface layer, the Monin-Obukhov similarity theory estimates the vertical turbulent diffusivity K_z in terms of the friction velocity u_*, and the Monin-Obukhov length L:

$$K_z = \frac{\kappa z u_*}{\phi_h} \tag{2}$$

where κ is the von Karman constant ($\kappa = 0.4$), z is the distance from the ground, and ϕ_h is the atmospheric stability function (e.g. Jacobson, 1999). Above the surface layer, the original form of the Monin-Obukhov similarity theory is no longer valid. In order to extend this theory to the entire boundary layer ($z/h < 1$) an evaluation of the Atmospheric Boundary Layer (ABL) height h is required. For this purpose, FALL3D 5.0 uses a simple parameterisation valid on the entire ABL (Ulke, 2000):

$$K_z = \begin{cases} \kappa u_* z \left(1 - \dfrac{z}{h}\right)\left(1 + 9.2\dfrac{h}{L}\dfrac{z}{h}\right)^{-1} & h/L \geq 0 \quad \text{stable} \\ \kappa u_* z \left(1 - \dfrac{z}{h}\right)\left(1 - 13\dfrac{h}{L}\dfrac{z}{h}\right)^{1/2} & h/L \leq 0 \quad \text{unstable} \end{cases} \tag{3}$$

Note that in the neutral case ($L \to \infty$) both expressions coincide. Finally, in the free atmosphere above the ABL ($z/h > 1$), K_z is considered a function of the local vertical wind gradient, a characteristic length scale l_c, and a stability function F_c depending on the Richardson number Ri:

$$K_z = l_c^2 \left|\frac{\partial V}{\partial z}\right| F_c(Ri) \tag{4}$$

For l_c and F_c the model adopts the relationship used by the CAM3 model (Collins et al., 2004) of the National Center for Atmospheric Research (NCAR):

$$l_c = \left(\frac{1}{\kappa z} + \frac{1}{\lambda_c} \right)^{-1} \tag{5}$$

$$F_c(Ri) = \begin{cases} \dfrac{1}{1+10Ri(1+8Ri)} & \text{stable} \quad (Ri > 0) \\ \sqrt{1-18Ri} & \text{unstable} \quad (Ri < 0) \end{cases} \tag{6}$$

where λ_c is the so-called asymptotic length scale ($\lambda_c \approx 30$m) while the Richardson number is calculated as $Ri = \dfrac{g}{\theta_v} \dfrac{\partial \theta_v / \partial z}{|\partial V / \partial z|^2}$ (with θ being virtual potential temperature).

On the other hand, for the horizontal eddy diffusivity $K_H = K_x = K_y$ FALL3D 5.0 assumes a large eddy parameterisation (Azad and Kitada, 1998):

$$K_H = \alpha \Delta x \Delta y \sqrt{\left(\frac{\partial v_x}{\partial y} + \frac{\partial v_y}{\partial x} \right)^2 + \frac{1}{2} \left[\left(\frac{\partial v_x}{\partial x} \right)^2 + \left(\frac{\partial v_y}{\partial y} \right)^2 \right]} \tag{7}$$

where α is a dimensionless constant of the order of unity that ranges from 0.1 to 5 depending on the size of the domain, and Δx and Δy are the horizontal grid spacings (commonly $\alpha = 0.5$). The model permits also to use other $K-$parametisations such as a constant value for K_z, a constant value for K_H or to estimate K_H using a Smagorinsky model as that used by RAMS model (for $\Delta z / \Delta \ll 1$ Pielke et al., 1992):

$$\begin{aligned} K_H &= R \max \left(K_{mh}; (C_{SH}\Delta)^2 \sqrt{\left(\frac{\partial v_x}{\partial y} + \frac{\partial v_y}{\partial x} \right)^2 + 2 \left[\left(\frac{\partial v_x}{\partial x} \right)^2 + \left(\frac{\partial v_y}{\partial y} \right)^2 \right]} \right) \\ K_{mh} &= 0.075 K_A \Delta^{4/3} \end{aligned} \tag{8}$$

where $\Delta = \sqrt{\Delta_x \Delta_y}$, C_{SH} is a dimensionless constant ranging from 0.135 to 0.32, K_A is a user defined parameter close to one, and $R \simeq 3$.

1.3 Settling velocity models

Settling velocity fits contemplated by FALL3D 5.0 include:

- `ARASTOOPOUR` (Arastoopour et al., 1982)

$$v_s = \sqrt{\frac{4gd\rho_p}{3C_d\rho_a}} \tag{9}$$

$$C_d = \begin{cases} \dfrac{24}{Re(1+15Re^{0.687})} & Re \leq 10^3 \\ 0.44 & Re > 10^3 \end{cases} \tag{10}$$

where ρ_a and ρ_p stand, respectivelly, for the air and particle densities, d is the particle diameter, Re is the Reynolds number and C_d is a drag coefficient.

- `GANSER` (Ganser , 1993)

$$v_s = \sqrt{\frac{4gd\rho_p}{3C_d\rho_a}} \tag{11}$$

$$\begin{cases} C_d = \dfrac{24}{ReK_1} \left\{ 1 + 0.1118 \left[Re \left(K_1 K_2 \right) \right]^{0.6567} \right\} + \dfrac{0.4305 K_2}{1 + \dfrac{0.4305}{ReK_1K_2}} \\ K_1 = 3/(1+2\psi^{-0.5}) \\ K_2 = 10^{1.84148(-\mathrm{Log}\psi)^{0.5743}} \end{cases} \tag{12}$$

where ψ is the particle sphericity (=1 for spherical particles).

Parameter	Scaling
Coordinates	$X = x \quad Y = y \quad Z = z - h(x,y)$
Velocities	$U = u_x \quad V = u_y \quad W = u_z J^{-1} \quad V_{sj} = v_{sj} J^{-1}$
Diffusion Coefficients	$K_X = K_x \quad K_Y = K_y \quad K_Z = K_z J^{-2}$
Concentration	$C = cJ$
Density	$\rho_* = \rho J$
Source Term	$S_* = SJ$

Table 1: Summary of the scaling factors for the terrain-following domain coordinate system ($x = X,\ y = Y,\ z \rightarrow Z$). J indicates the Jacobian of the coordinate system transformation.

- `WILSON` (Wilson and Wang , 1979)

$$v_s = \sqrt{\frac{4gd\rho_p}{3C_d\rho_a}} \tag{13}$$

$$C_d = \begin{cases} \dfrac{24}{Re}\phi^{-0.828} + 2\sqrt{1.07-\phi} & Re \leq 10^2 \\ \dfrac{1 - C_d(Re=100)}{900}(Re - 1000) + 1 & 10^2 \leq Re \leq 10^3 \\ 1 & Re \geq 10^3 \end{cases} \tag{14}$$

  where ϕ is the aspect ratio $(b+c)/2a$ (a, b, c ellipsoidal semi-axes).

- `DELLINO` (Dellino et al., 2005).

$$v_s = \frac{1.2065\mu}{d\rho_a}\left(d^3(\rho_p - \rho_a)\rho_a \Psi^{1.6}/\mu^2\right)^{0.5206} \tag{15}$$

  where μ is the air viscosity and Ψ is the shape factor (sphericity to circularity ratio).

2 Program setup

2.1 Installation

- On a Windows OS download and decompress the file `Fall3d-5.0.tar.gz` on your selected directory. The `Fall3d-5.0.tar` file already contains Windows executables for FALL3D 5.0 and the rest of included utility programs. It is not strictly necessary to have a FORTRAN90 compiler. The untaring of `Fall3d-5.0.tar` will create the folders described in the Table 2.

- On a Unix/Linux/Mac X operating system:

  1. Decompress and then untar the file `Fall3d-5.0.tar` issuing the command "`tar xvf Fall3d-5.0.tar`". This will generate directory `Fall3d-5.0` (see Table 2).
  2. Compile the program FALL3D 5.0 . Enter the directory `Fall3d-5.0/Sources`, then issue the command "`make`" to produce the executable Fall3d.exe. You can edit the Makefile to select your favourite compiler. After compilation you may issue the command "`make clean`" to remove unneeded files. If you are not going to run on a Windows platform you can also delete the executable Fall3d.win.exe.
  3. Compile the (optional) utility program SETGRN . Enter the directory `Fall3d-5.0/Utilities/SetGrn/Sources`, then issue the command "`make`" to produce the executable SerGrn.exe. You can edit the Makefile to select your favourite compiler. After compilation you may issue the command "`make clean`" to remove unneeded files. If you are not going to run on a Windows platform you can also delete the executable SetGrn.win.exe.
  4. Compile the (optional) utility program SETSRC . Enter the directory `Fall3-3.0/Utilities/SetSrc/Sources`, then issue the command "`make`" to produce the executable SerSrc.exe. You can edit the Makefile to select your favourite compiler. After compilation you may issue the command "`make clean`" to remove unneeded files. If you are not going to run on a Windows platform you can also delete the executable SetSrc.win.exe.
  5. Compile the (optional) utility program BUILDDBS . Enter the directory `Fall3-3.0/Utilities/BuildDbs/Sources`, then issue the command "`make`" to produce the executable SerSrc.exe. You can edit the Makefile to select your favourite compiler. After compilation you may issue the command "`make clean`" to remove unneeded files. If you are not going to run on a Windows platform you can also delete the executable BuildDbs.win.exe.
  6. Compile the (optional) utility program MODELPOSTP . Enter the directory `Fazmap-3.0/Utilities/ModelPostp/Sources`, then issue the command "`make`" to produce the executable ModelPostp.exe. You can edit the Makefile to select your favourite compiler. After compilation you may issue the command "`make clean`" to remove unneeded files. If you are not going to run on a Windows platform you can also delete the executable ModelPostp.win.exe.

2.2 Folder structure

Table 2 shows the folder structure. The directory `Fall3d-5.0/Sources` contains the FALL3D 5.0 source files, the directory `Fall3d-5.0/Utilities` contains the programs SETGRN , SETSRC , BUILDDBS , and MODELPOSTP and, finally, the directory `Fall3d-5.0/Runs` contains the runs, one within each own folder. An example run named "Example" is provided with the installation.

2.3 Program run

FALL3D 5.0 can be launched typing
"`Fall3d.exe FileInp FileSrc FileGrn FileDbs FileLst FileRes`" (on a Unix/Linux/Mac X OS)
or
"`Fall3d.win.exe FileInp FileSrc FileGrn FileDbs FileLst FileRes`" (on a Windows OS), where

- `FileInp`: Name (including path) of the control input file (see section 3.1).
- `FileSrc`: Name (including path) of the source input file (see section 3.2).

Table 2: Default directory structure.

Fall3d-5.0	Sources	
	Utilities	SetGrn
		SetSrc
		BuildDbs
		ModelPostp
	Runs	Example
		....

- `FileGrn`: Name (including path) of the granulometry input file (see section 3.3).
- `FileDbs`: Name (including path) of the meteo data input file (see section 3.4).
- `FileLst`: Name (including path) of the output list file (see section 4.1).
- `FileInp`: Name (including path) of the FALL3D 5.0 results file (see section 4.2).

Note that filenames (and locations) are passed as a program call argument. It is highly recomended to launch FALL3D 5.0 through the script files included in the distribution.

- On a Windows OS enter the folder `Fall3d-5.0`, edit the script `Script-Fall3d-Win.bat` to change the "problemname" variable and launch the script.
- On a Mac X/Unix/Linux OS enter the folder `Fall3d-5.0`, edit the script `Script-Fall3d-Unix` to change the "problemname" variable and launch the script.

NOTE: To create a new run you can simply create a new folder, copy the control input file of the example (Example.inp) and modifiy the script line which defines the "problemname" variable.

3 The FALL3D 5.0 input files

3.1 The control file `FileInp`

The FALL3D 5.0 control file is passed to the program as a call argument. This file is made up with a set of blocks that define all the computational and physical parameters needed by the dispersion model (Table 3 shows an example of control file). Parameters within a block are listed one per record, in arbitrary order, and can optionally be followed by one or more blank spaces and a comment. A detailed description of each record is given below. Real numbers can be expressed following the FORTRAN notation (*e.g.*, $12e7 = 12 \times 10^7$).

3.1.1 BLOCK TIME_UTC

This block of data defines variables related to time.

- `YEAR`: Simulation year.
- `MONTH`: Simulation month (1-12).
- `DAY`: Simulation day (1-31).
- `RUN_START_(HOURS_AFTER_00)`: Run start hour (after 0000UTC).
- `ERUPTION_END_(HOURS_AFTER_00)` : Eruption end hour (after 0000UTC). If the SETSRC program is used to generate the source term, this is the time instant at which source is switched off.
- `RUN_END_(HOURS_AFTER_00)`: Run end hour (after 0000UTC). Note that, in general, a run should continue even when the source term is switched off (*i.e.* when the eruption has stop) in order to give time for the remaining airborne particles to fall.

3.1.2 BLOCK FALL3D

This block of data defines the rest of variables needed by the program.

- `ZLAYER_(M)`: Heights (in m) of the z-layers in terrain following coordinates, *i.e.* above the vent. It is not necessary to specify the number of vertical layers since it is automatically calculated by the program.
- `TERMINAL_VELOCITY_MODEL`: Type of terminal settling velocity model. Possibilities are `ARASTOOPOUR` (Arastoopour et al., 1982), `GANSER` (Ganser , 1993), `WILSON` (Wilson and Wang , 1979) and `DELLINO` (Dellino et al., 2005).
- `TERMINAL_VELOCITY_MODEL_FACTOR`: Model dependent factor. For `ARASTOOPOUR` it is not used. For `GANSER` it is the sphericity ψ (see 12). For `WILSON` it is the aspect ratio ϕ (see 14). Finally, for `DELLINO` it is the shape factor Ψ (see 15).
- `VERTICAL_TURBULENCE_MODEL`: Type of model for vertical diffusion. Possibilities are `CONSTANT` or `SIMILARITY`. See Costa et al. (2005) for details.
- `VERTICAL_DIFFUSION_COEFFICIENT_(M2/S)`: Value of the diffusion coefficient (in m^2/s). Only used if `VERTICAL_TURBULENCE_MODEL = CONSTANT`
- `HORIZONTAL_TURBULENCE_MODEL`: Type of model for horizontal diffusion. Possibilities are `CONSTANT`, `PIELKE`, or `RAMS`. See Costa et al. (2005) for details.
- `HORIZONTAL_DIFFUSION_COEFFICIENT_(M2/S)`: Value of the diffusion coefficient (in m^2/s). Only used if `HORIZONTAL_TURBULENCE_MODEL = CONSTANT`.
- `SAFETY_FACTOR`: Safety factor. This is a factor that multiplies the critical time step. It should be equal or lower than 1.0 (typically 0.8-0.9) to ensure stability.
- `USE_LIMITER_METHOD`: Flag to indicate the use of limiter. Possibilities are `YES` or `NO`. If `YES` the algoritm uses a limiter method. See Costa et al. (2005) for details.

Table 3: Sample of the input control file `FileInp`.

```
TIME_UTC
YEAR = 2007
MONTH = 03
DAY = 01
RUN_START_(HOURS_AFTER_00) = 0.
RUN_END_(HOURS_AFTER_00) = 3.
ERUPTION_END_(HOURS_AFTER_00) = 3.
RUN_END_(HOURS_AFTER_00) = 6.

FALL3D
ZLAYER_(M) = 0 500 1000 1500 2000 2500 3000 3500 4000 4500 5000
TERMINAL_VELOCITY_MODEL = GANSER
TERMINAL_VELOCITY_MODEL_FACTOR = 0.8
VERTICAL_TURBULENCE_MODEL = SIMILARITY
VERTICAL_DIFFUSION_COEFFICIENT_(M2/S) = 100.
HORIZONTAL_TURBULENCE_MODEL = PIELKE
HORIZONTAL_DIFFUSION_COEFFICIENT_(M2/S) = 1000.
SAFETY_FACTOR = 0.9
USE_LIMITER_METHOD = yes
POSTPROCESS_TIME_INTERVAL_(HOURS) = 2.
```

- `POSTPROCESS_TIME_INTERVAL_(HOURS)`: Time interval to output results (in h). Results are also output at the end of the run.

3.2 The source file `FileSrc`

The FALL3D 5.0 source file is an ASCII file containing the definition of the source term. The source is defined at time intervals during which source values are kept constant. The number, position and values (*i.e.* Mass Flow Rate) of the source points can, however, vary from one time slice to another. There is no restriction on the number and duration of the time intervals. It allows, in practise, to discretize any type of source term. This file can be created by the utility program SETSRC . The file format is described in Table 4 and the meaning of the used symbols is the following:

- `itime1`: Starting time (in sec after 00UTC) of the time slice.
- `itime2`: End time (in sec after 00UTC) of the time slice.
- `nsrc`: Number of source points (can vary from one interval to another).
- `nc`: Number of granulometric classes.
- `MFR`: Mass flow rate (in kg/s).
- `x`: x-coordinate of the source isrc (UTM coordinates in m).
- `y`: y-coordinate of the source isrc (UTM coordinates in m).
- `z`: z-coordinate of the source isrc (terrain following coordinates in m, *i.e.* above the vent).
- `src`: Mass flow rate (in kg/s) of each granulometric class for this point source. It must be verified that $\sum\sum src(isrc, ic) = MFR$.

Table 4: Format of the source file `FileSrc`. Repeat this block for each time slice

.

itime1 itime2
nsrc nc
MFR
x y z src(1,1) ... src(1,nc)
...
x y z src(nsrc,1) ... src(nsrc,nc)

Table 5: Format of the granulometry file `FileGrn`.

nc
diam(1) rho(1) fc(1)
...
diam(nc) rho(nc) fc(nc)

3.3 The granulometry file `FileGrn`

The FALL3D 5.0 granulometry file is an ASCII file containing particle densities and granulometric distribution. This file can be created by the utility program SETGRN . The file format is described in Table 5 and the meaning of the used symbols is the following:

- `nc`: Number of granulometric classes.
- `diam`: Class diameter (in mm).
- `rho`: Class density (in kg/m^3).
- `fc`: Class mass fraction (0-1). If must verify that $\sum$ fc $= 1$.

3.4 The source file `FileDbs`

The FALL3D 5.0 DataBase file is a binary file created by the BUILDDBS utility program. It contains the meteorological database stored in a direct access binary file. In addition to it, BUILDDBS creates also an ASCII file that contains the explanation of each record of the FileDbs.

4 The FALL3D 5.0 output files

4.1 The list file `FileLst`

This file contains information concerning the run (summary of input data, run time error messages, CPU time, *etc.*).

4.2 The results file `FileRes`

This is a binary file with the results from a FALL3D 5.0 run. This file must be processed by the MODELPOSTP utility to produce "human readeable" files (normally in GRD format).

5 The utility programs

5.1 The program SETGRN

The granulometric distribution is defined in the granulometry file (see section 3.3). The program SETGRN is an utility that reads the `GRANULOMETRY` block from the control input file and generates the granulometry

file assuming that the mass fraction of particles follows either a linear or Gaussian distribution in Φ and that the density of particles varies linearly with Φ. Note that other granulometric distributions different from a linear/Gaussian can also be considered. However, in this case, the FALL3D 5.0 granulometry file can not be generated by SETGRN but must be supplied directly by the user.

5.1.1 Program execution

To run SETGRN (see section 2.1 for installation details) simply type
"`SetGrn.exe FileLog FileInp FileGrn`" (on a Unix/Linux/Mac X OS) or
"`SetGrn.win.exe FileLog FileInp FileGrn`" (on a Windows OS) where

- `FileLog`: Name (including path) of the SETGRN output log file.
- `FileInp`: Name (including path) of the control input file that contains the `GRANULOMETRY` block. Normally this file coincides with the FALL3D 5.0 input file.
- `FileGrn`: Name (including path) of the granulometry file. This is the output from SETGRN that is used later by FALL3D 5.0 as input.

Note that filenames are passed as a program call argument. It is highly recomended to launch SETGRN through the script files included in the distribution.

- On a Windows OS go to the folder `Fall3d-5.0/Utilities/SetGrn`, edit the script `Script-SetGrn-Win.bat` to change the "problemname" variable and launch the script.
- On a Mac X/Unix/Linux OS enter the folder `Fall3d-5.0/Utilities/SetGrn`, edit the script `Script-SetGrn-Unix` to change the "problemname" variable and launch the script.

5.1.2 The GRANULOMETRY block

This block of data (see Table 6) defines the variables needed by SETGRN . Commonly this block is appended to the FALL3D 5.0 control input file. The meaning of each record is the following:

- `NUMBER_OF_CLASSES`: Number of granulometric classes.
- `DISTRIBUTION`: Type of distribution. It can be `LINEAR` or `GAUSSIAN`.
- `FI_MIN`: Minimum value of Φ.
- `FI_MAX`: Maximum value of Φ.
- `FI_MEAN`: Mean value of Φ. Only used if (`DISTRIBUTION = GAUSSIAN`).
- `FI_DISP`: Value of σ in the Gaussian distribution. Only used if (`DISTRIBUTION = GAUSSIAN`).
- `FI_SLOPE`: Slope of the linear distribution. Only used if (`DISTRIBUTION = LINEAR`).
- `MINIMUM_DENSITY`: Minimium value of density (in kg/m^3) and associated value of Φ. For values of Φ lower than this value (larger particles) density is assumed to be constant and equal to the minimum value.
- `MAXIMUM_DENSITY`: Maximum value of density (in kg/m^3) and associated value of Φ. For values of Φ larger than this value (smaller particles) density is assumed to be constant and equal to the maximum value.

Table 6: Sample of the SETGRN input file. Normally this block is appended at the end of the input file.

```
GRANULOMETRY
NUMBER_OF_CLASSES = 12
DISTRIBUTION = GAUSSIAN
FI_MIN = 0
FI_MAX = 5
GAUSSIAN_DISTRIBUTION
FI_MEAN = 2.5
FI_DISP = 1.5
LINEAR_DISTRIBUTION
FI_SLOPE = 0.5
MINIMUM_DENSITY = 1080.  -3.
MAXIMUM_DENSITY = 2300 5.
```

5.2 The program SETSRC

The distribution of sources is defined in a source file (see section 3.2). The program SETSRC is an utility that reads the SOURCE block from the control input file and generates a source file. The source term is constant for a given time interval but there is no limit on the number and duration of the time intervals. It allows, in practise, to discretize any kind of time-dependency (time-dependent mass flow rate, column height, etc.). The program admits three possibilities: point source (mass is released in a single source point), Suzuki distribution (Suzuki , 1983; Pfeiffer et al., 2005), and buoyant plume model (Bursik , 2001). The last option is more elaborated and involves the solution of the 1D radial-averaged plume governing equations that describe the convective region of an eruptive column. These equations are intimately coupled with the wind field which, for small to medium size plumes, may induce a substantial plume bent-over and subsequent variations of plume height and mass release location. For this reason, when this option switched on, the program reads the values of the wind field from a meteorological file, computes the averaged wind direction and solves the plume governing equations for each time interval and particle class accounting for wind. Note that it introduces a time dependence in the source term even when all the eruptive parameters (mass flow rate, class fraction, etc.) are kept constant in time.

5.2.1 Program execution

To run SETSRC (see section 2.1 for installation details) simply type
"`SetSrc.exe FileLog FileInp FileSrc FileGrn FileMet MODEL MESH`" (on a Unix/Linux/Mac X OS)
or
"`SetSrc.win.exe FileLog FileInp FileSrc FileGrn FileMet MODEL MESH`" (on a Windows OS) where

- `FileLog`: Name (including path) of the SETSRC output log file.
- `FileInp`: Name (including path) of the control input file that contains the SOURCE block. Normally this file coincides with the FALL3D 5.0 input file.
- `FileSrc`: Name (including path) of the source file. This is the output from SETSRC that is used later by FALL3D 5.0 as input.
- `FileGrn`: Name (including path) of the granulometry file (normally generated previously by SETGRN).
- `FileMet`: Name (including path) of the meteo data file generated by BUILDDBS .
- `MODEL`: Flag indicating the name of the model. For FALL3D 5.0 it is always "FALL3D".
- `MESH`: Flag that indicates if the discrete source points must be interpolated onto a mesh. For FALL3D 5.0 it is always "YES".

Note that filenames are passed as a program call argument. It is highly recomended to launch SETSRC through the script files included in the distribution.

Table 7: Sample of the SETSRC input file. Normally this block is appended at the end of the FALL3D 5.0 input file. In this example a Suzuki source and two time intervals are assumed.

```
SOURCE
X_VENT_(UTM_M) = 500080.
Y_VENT_(UTM_M) = 4177690.
MASS_FLOW_RATE_(KGS) = 5d4 4d4          (One value for each source time interval)
SOURCE_TYPE = SUZUKI
POINT_SOURCE                            (Variables below are used only if SOURCE_TYPE = POINT)
HEIGHT_ABOVE_VENT_(M) = 2000.
SUZUKI_SOURCE                           (Variables below are used only if SOURCE_TYPE = SUZUKI)
HEIGHT_ABOVE_VENT_(M) = 3000.  2800.    (One value for each source time interval)
A = 4.  4.                              (One value for each source time interval)
L = 5.  5.                              (One value for each source time interval)
PLUME_SOURCE                            (Variables below are used only if SOURCE_TYPE = PLUME)
EXIT_VELOCIY_(MS) = 100.
EXIT_TEMPERATURE_(K) = 1073.
EXIT_VOLATILE_FRACTION_(IN%) = 0.
```

- On a Windows OS go to the folder `Fall3d-5.0/Utilities/SetSrc`, edit the script `Script-SetSrc-Win.bat` to change the "problemname" variable and launch the script.
- On a Mac X/Unix/Linux OS enter the folder `Fall3d-5.0/Utilities/SetSrc`, edit the script `Script-SetSrc-Unix` to change the "problemname" variable and launch the script.

5.2.2 The SOURCE block

This block of data (see Table 7) defines the variables needed by SETSRC . Commonly this block is appended to the FALL3D 5.0 control input file. The meaning of each record is the following:

- `X_VENT_(UTM_M)`: x-coordinate of the vent (UTM coordinates in m).
- `Y_VENT_(UTM_M)`: y-coordinate of the vent (UTM coordinates in m).
- `MASS_FLOW_RATE_(KGS)`: Values of the mass flow rate (in kg/s). One value for each time interval. The duration of each time interval is constant and given by `RUN_START_(HOURS_AFTER_00)` minus `ERUPTION_END_(HOURS_AFTER_00)` divided by the number of time intervals (automatically computed by the program from the number of values).
- `SOURCE_TYPE`: Type of source distribution. Possibilities are `POINT`, `SUZUKI` or `PLUME`.
- `HEIGHT_ABOVE_VENT_(M)`: Heights of the plume (in m above the vent). One value for each time interval.
- `A`: Parameter A in the Suzuki distribution. One value for each time interval. Used only if `SOURCE_TYPE = SUZUKI`.
- `L`: Parameter L in the Suzuki distribution. One value for each time interval. Used only if `SOURCE_TYPE = SUZUKI`.
- `EXIT_VELOCIY_(MS)`: Magma exit velocity (in m/s) at the vent. One value for each time interval. Used only if `SOURCE_TYPE = PLUME`.
- `EXIT_TEMPERATURE_(K)`: Magma exit temperature (in ^oK) at the vent. One value for each time interval. Used only if `SOURCE_TYPE = PLUME`.
- `EXIT_VOLATILE_FRACTION_(IN%)`: Magma volatile mass fraction at the vent. One value for each time interval. Used only if `SOURCE_TYPE = PLUME`.

5.3 The program BUILDDBS

This program generates the database files for FALL3D 5.0 using as input either a veritcal profile (sounding) plus a topography file (in format GRD) or an output of the meteorological processor CALMET (version 6). The latter option is prefereable because CALMET generates a 3D wind field that accounts for topographic effects and determines values for micrometeorological variables in the Atmospheric Boundary Layer (ABL). In the horizontal, and the database use the same spatial discretization.

5.3.1 Program execution

To run BUILDDBS (see section 2.1 for installation details) simply type
"`BuildDbs.exe FileLog FileInp FileDat FileDbs FileLst TypeData FileTop`" (on a Unix/Linux/Mac X OS) or
"`BuildDbs.win.exe FileLog FileInp FileDat FileDbs FileLst TypeData FileTop`"(on a Windows OS) where

- `FileLog`: Name (including path) of the BUILDDBS output log file.
- `FileInp`: Name (including path) of the control input file that contains the `METEO_DATABASE` block. Normally this file coincides with the FALL3D 5.0 input file.
- `FileDat`: Name (including path) of the meteo data file. This is either the vertical profile file or the binary output from CALMET (version 6) depending on the value of `TypeData`
- `FileDbs`: Name (including path) of the DataBase file. This is the output from BUILDDBS that is used later by FALL3D 5.0 as input.
- `FileLst`: Name (including path) of the DataBase record descriptor file.
- `TypeData`: Flag to indicate the origin of meteorological data. Possibilities are `PROFILE` or `CALMET62`.
- `FileTop`: Name (including path) of the GRD topography file.

Note that filenames are passed as a program call argument. It is highly recomended to launch BUILDDBS through the script files included in the distribution.

- On a Windows OS go to the folder `Fall3d-5.0/Utilities/BuildDbs`, edit the script `Script-BuildDbs-Win.bat` to change the "problemname" variable and launch the script.
- On a Mac X/Unix/Linux OS enter the folder `Fall3d-5.0/Utilities/BuildDbs`, edit the script `Script-BuildDbs-Unix` to change the "problemname" variable and launch the script.

5.3.2 The METEO_DATABASE block

This block of data (see Table 8) defines the variables needed by BUILDDBS . Commonly this block is appended to the FALL3D 5.0 control input file. The meaning of each record is the following:

- `YEAR`: Simulation year.
- `MONTH`: Simulation month (1-12).
- `DAY`: Simulation day (1-31).
- `BEGIN_METEO_DATA_(HOURS_AFTER_00)`: Time (in h after 0000UTC) at which meteorological data start.
- `END_METEO_DATA_(HOURS_AFTER_00)`: Time (in h after 0000UTC) at which meteorological data ends. The meteo time slice should include the simulation time slice defined by the records `RUN_START_(HOURS_AFTER_00)` and `RUN_END_(HOURS_AFTER_00)` of the `TIME_UTC` block.
- `TIME_STEP_METEO_DATA_(MIN)`: Time step (in min) of the meteo data.
- `X_ORIGIN_(UTM_M)`: x-coordinate of the grid (UTM coordinates in m).

Table 8: Sample of the BUILDDBS input file. Normally this block is appended at the end of the FALL3D 5.0 input file.

```
METEO_DATABASE
YEAR = 2007
MONTH = 03
DAY = 01
BEGIN_METEO_DATA_(HOURS_AFTER_00) = 0
END_METEO_DATA_(HOURS_AFTER_00) = 6
TIME_STEP_METEO_DATA_(MIN) = 60
X_ORIGIN_(UTM_M) = 450000
Y_ORIGIN_(UTM_M) = 4125000
CELL_SIZE_(KM) = 2.0
NX = 51
NY = 51
Z_LAYER_(M) = 0.  10.  40.  100.  250.  500.  1000.  2500.  5000.  7500.  10000.
```

- `Y_ORIGIN_(UTM_M)`: y-coordinate of the grid (UTM coordinates in m).
- `CELL_SIZE_(KM)`: Horizontal spatial discretization (in km).
- `NX`: Number of grid cells in the x-direction (for both FALL3D 5.0 and BUILDDBS).
- `NY`: Number of grid cells in the y-direction (for both FALL3D 5.0 and BUILDDBS).
- `ZLAYER_(M)`: Heights (in m) of the database z-layers. If `TypeData` is `PROFILE` then BUILDDBS interpolates the measured values of velocity and temperature at these heights. If `TypeData` is `CALMET62` the heights represent the CALMET cell faces. Note that the vertical discretizations of BUILDDBS and FALL3D 5.0 can differ. The latter is defined in the record `ZLAYER_(M)` of the `FALL3D` block.

5.3.3 The meteo data file `FileDat`

If `TYPE_OF_DATA = CALMET` this is the (binary) calmet output file. On the contrary, if `TYPE_OF_DATA = PROFILE` this is an ASCII file containing the definition of the vertical wind profile. In this case, the file format is described in Table 9 and the meaning of the used symbols is the following:

- `itime1`: Starting time (in sec after 00UTC) of the meteo data time slice.
- `itime2`: End time (in sec after 00UTC) of the meteo data time slice.
- `nz`: Number of vertical layers.
- `z`: Vertical coordinate of the layer (in m, terrain following).
- `ux`: wind x-velocity (in m/s).
- `uy`: wind y-velocity (in m/s).
- `T`: Air temperature (in ^oC).

5.3.4 The topography file `FileTop`

This is a file in GRD format containing the topography. It is used BUILDDBS only when `TYPE_OF_DATA = PROFILE` The file format is described in Table 10 and the meaning of the used symbols is the following:

- `nx`: Number of cells in the x-direction. Coincides with the value defined in the `NX` record of the `METEO_DATABASE` block .

Table 9: Format of the meteo data file FileDat for the `PROFILE` case. Repeat this block for each meteo time increment.

itime1 itime2
nz
z(1) ux(1) ux(1) T(1)
...
z(nz) ux(nz) ux(nz) T(nz)

Table 10: Format of the topography file FileTop for the `PROFILE` case.

DSAA
nx ny
xo xf
yo yf
zmin zmax
z(1,1) ... z(1,nx)
...
z(ny,1) ... z(ny,nx)

- `ny`: Number of cells in the y-direction. Coincides with the value defined in the `NY` record of the `METEO_DATABASE` block .
- `xo`: x-coordinate of the grid bottom left corner (UTM coordinates in m). Coincides with the value defined in the `X_ORIGIN_(UTM_M)` record of the `METEO_DATABASE` block .
- `xf`: x-coordinate of the grid top right corner (UTM coordinates in m).
- `yo`: y-coordinate of the grid bottom left corner (UTM coordinates in m). Coincides with the value defined in the `Y_ORIGIN_(UTM_M)` record of the `METEO_DATABASE` block .
- `yf`: y-coordinate of the grid top right corner (UTM coordinates in m).
- `zmin`: Minimum value of z in the domain.
- `zmax`: Maximum value of z in the domain.
- `z`: Height (in m) of each grid point.

5.4 The program ModelPostp

The program ModelPostp (alias for Model Postprocess) is an optional utility that reads the output binary file of Fall3d 5.0 , calculates some relevant quantities at selected time instants and produces elementary maps in GRD and PS formats.

5.4.1 Program execution

To run ModelPostp (see section 2.1 for installation details) simply type
"`ModelPostp.exe FileLog FileInp FileRes BASERES`" (on a Unix/Linux/Mac X OS) or
"`ModelPostp.win.exe FileLog FileInp FileRes BASERES`" (on a Windows OS) where

- `FileLog`: Name (including path) of the ModelPostp output log file.
- `FileInp`: Name (including path) of the control input file that contains the `POSTPROCESS_MODELS` block. Normally this file coincides with the Fall3d 5.0 input file.

- `FileRes`: Name (including path) of the FALL3D 5.0 results file. This is the output from FALL3D 5.0 that is used by as input.
- `BASERES`: Path where the MODELPOSTP output files are dump.

Note that filenames are passed as a program call argument. It is highly recomended to launch MODELPOSTP through the script files included in the distribution.

- On a Windows OS go to the folder `Fall3d-5.0/Utilities/ModelPostp`, edit the script `Script-ModelPostp-Win.bat` to change the "problemname" variable and launch the script.
- On a Mac X/Unix/Linux OS enter the folder `Fall3d-5.0/Utilities/ModelPostp`, edit the script `Script-ModelPostp-Unix` to change the "problemname" variable and launch the script.

5.4.2 The POSTPROCESS_MODELS block

This block of data (see Table 11) defines the variables needed by MODELPOSTP . Commonly this block is appended to the FALL3D 5.0 control input file. The meaning of each record is the following:

- `OUTPUT_FILES_IN_GRD_FORMAT`: Possibilities are YES or NO. If YES, MODELPOSTP plots files in GRD format. Files in GRD format can be readed directly by several plotting programs like the commercial software GRAPHER. Alternativelly, the user may also generate its own plots using functons from several free packages (*e.g.* gnuplot in FORTRAN).
- `OUTPUT_FILES_IN_PS_FORMAT`: Possibilities are YES or NO. If YES, MODELPOSTP plots files in PS format.
- `MAP_TOTAL_LOAD`: Possibilities are YES or NO. If YES MODELPOSTP plots the total ground load.
- `UNITS`: Units of `MAP_TOTAL_LOAD`. It must be KG/M2.
- `CONTOUR_LEVELS`: Values of the contour levels for `MAP_TOTAL_LOAD`. Only used when `OUTPUT_FILES_IN_PS_FORMAT` is YES.
- `MAP_CLASS_LOAD`: Possibilities are YES or NO. If YES MODELPOSTP plots the class ground load.
- `UNITS`: Units of `MAP_CLASS_LOAD`. It must be KG/M2.
- `CONTOUR_LEVELS`: Values of the contour levels for `MAP_CLASS_LOAD`. Only used when `OUTPUT_FILES_IN_PS_FORMAT` is YES.
- `MAP_DEPOSIT_THICKNESS`: Possibilities are YES or NO. If YES MODELPOSTP plots total deposit thickness.
- `UNITS`: Units of `MAP_DEPOSIT_THICKNESS`. Possibilities are MM (for mm), CM (for cm), and M (for m).
- `COMPACTATION_FACTOR`: Deposit compactation factor.
- `CONTOUR_LEVELS`: Values of the contour levels for `MAP_DEPOSIT_THICKNESS`. Only used when `OUTPUT_FILES_IN_PS_FORMAT` is YES.
- `MAP_TOTAL_CONCENTRATION`: Possibilities are YES or NO. If YES MODELPOSTP plots the total concentration at certain z-levels.
- `UNITS`: Units of `MAP_TOTAL_CONCENTRATION`. It must be KG/M3.
- `Z_CUTS_(M)`: z-coordinates of the layers at which concentration is output.
- `CONTOUR_LEVELS`: Values of the contour levels for `MAP_TOTAL_CONCENTRATION`. Only used when `OUTPUT_FILES_IN_PS_FORMAT` is YES.
- `MAP_Z_CUMMULATIVE_CONCENTRATION`: Possibilities are YES or NO. If YES MODELPOSTP plots the z cummulative concentration (vertical integration).

Table 11: Sample of the ModelPostp input file. Normally this block is appended at the end of the input file.

```
POSTPROCESS_MODELS
OUTPUT_FILES_IN_GRD_FORMAT = YES
OUTPUT_FILES_IN_PS_FORMAT = NO
MAP_TOTAL_LOAD = YES
UNITS = KG/M2
CONTOUR_LEVELS = 0.1 0.25 0.5 1.  5.  10.  50.      (Only used if OUTPUT_FILES_IN_PS_FORMAT=YES)
MAP_CLASS_LOAD = NO
UNITS = KG/M2
CONTOUR_LEVELS = 0.1 0.25 0.5 1.  5.  10.  50.      (Only used if OUTPUT_FILES_IN_PS_FORMAT=YES)
MAP_DEPOSIT_THICKNESS = NO
UNITS = MM
COMPACTATION_FACTOR = 0.7
CONTOUR_LEVELS = 0.1 1.  5.  10.  50.  100.  500.  (Only used if OUTPUT_FILES_IN_PS_FORMAT=YES)
MAP_TOTAL_CONCENTRATION = YES
UNITS = KG/M3
Z_CUTS_(M) = 1000.  2000.
CONTOUR_LEVELS = 1e-5 1e-4                          (Only used if OUTPUT_FILES_IN_PS_FORMAT=YES)
MAP_Z_CUMMULATIVE_CONCENTRATION = YES
UNITS = KG/M2
CONTOUR_LEVELS = 0.01 0.1 1.  10.                   (Only used if OUTPUT_FILES_IN_PS_FORMAT=YES)
MAP_Z_MAXIMUM_CONCENTRATION = YES
UNITS = KG/M3
CONTOUR_LEVELS = 1e-4 1e-3                          (Only used if OUTPUT_FILES_IN_PS_FORMAT=YES)
```

- UNITS: Units of MAP_Z_CUMMULATIVE_CONCENTRATION. It must be KG/M2.
- CONTOUR_LEVELS: Values of the contour levels for MAP_Z_CUMMULATIVE_CONCENTRATION. Only used when
  OUTPUT_FILES_IN_PS_FORMAT is YES.
- MAP_Z_MAXIMUM_CONCENTRATION: Possibilities are YES or NO. If YES ModelPostp plots the maximum value of concentration along the vertical for each point. This variable can be useful for flight safety concentration tresholds.
- UNITS: Units of MAP_Z_MAXIMUM_CONCENTRATION. It must be KG/M3.
- CONTOUR_LEVELS: Values of the contour levels for MAP_Z_MAXIMUM_CONCENTRATION. Only used when OUTPUT_FILES_IN_PS_FORMAT is YES.

6 References

Arastoopour, H., Wang, C., Weil, S., 1982. Particle-particle interaction force in a dilute gassolid system. Chemical Engineering Science 37, 1379–1386.

Azad, A., Kitada, T., 1998. Characteristic of the air pollution in the city of Dhaka, Bangladesh in winter. Atmos. Environ. 32, 1991–2005.

Bursik, M., 2001. Effect of wind on the rise height of volcanic plumes. Geophys. Res. Lett. 18, 3621-3624.

Collins, W., Rasch, P., Boville, B., Hack, J., McCaa, J., Williamson, D., Kiehl, J., Briegleb, B., 2004. Description of the NCAR Community Atmosphere Model (CAM 3.0). Technical Report NCAR/TN-464+STR, National Center for Atmospheric Research, Boulder, Colorado.

Costa, A., Macedonio, G., Folch, A., 2006. A three-dimensional Eulerian model for transport and deposition of volcanic ashes. Earth Planet. Sci. Lett., 241 (34), 634–647.

Dellino, P., D. Mele, R. Bonasia, G. Braia, L. La Volpe, R. Sulpizio, 2005. The analysis of the influence of pumice shape on its terminal velocity, Geophys. Res. Lett., 32, L21306.

Ganser, G., 1993. A rational approachto drag prediction spherical and nonspherical particles. Powder Technology 77, 143–152.

Jacobson, M., 1999. Fundamentals of atmospheric modelling, 1st Edition. Cambridge University Press, New York.

Pfeiffer, T., Costa, A., Macedonio, G., 2005. A model for the numerical simulation of tephra fall deposits. J. Volcanol. Geotherm. Res. 140, 273–294.

Pielke, R., Cotton, W., Walko, R., Tremback, C., Nicholls, M., Moran, M., Wesley, D., Lee, T., Copeland, J., 1992. A comprehensive meteorological modeling system-RAMS. Meteor. Atmos. Phys. 49, 69–91.

Ulke, A., 2000. New turbulent parameterization for a dispersion model in atmospheric boundary layer. Atmos. Environ. 34, 1029–1042.

Wilson, L., and T. C. Huang, 1979. The influence of shape on the atmospheric settling velocity of volcanic ash particles, Earth Planet. Sci. Lett. 44, 311–324.

Suzuki, T., 1983. A theoretical model for dispersion of tephra. In: D. Shimozuru, I. Yokoyama (Eds.), Arc Volcanism: Physics and Tectonics, Terra Scientific Publishing Company (TERRAPUB), Tokyo.